AMENDMENTS TO THE CLAIMS

Please amend the claims as indicated below. The language being added is underlined ("__") and the language being deleted contains a strikethrough ("—").

LISTING OF CLAIMS

1.   (Previously Presented) A data converter for converting a group of vectors from a time serial to a time parallel format, wherein in the time serial format, sets of corresponding components of the vectors each have a time slot, and in time parallel format, each vector has a time slot, the converter comprising:

an input rotator configured to rotate each set of corresponding components of all time serial vectors by an amount that depends on the time slot of the set of corresponding components;

a bank of register files coupled to the input rotator to receive the rotated set of corresponding components, and having a register file in the bank configured to store each rotated set of corresponding components;

an output rotator coupled to the bank of registers files, for receiving and rotating the components of a vector an amount that depends on the time slot of the vector to generate a vector having time parallel format; and

a controller configured to control the addressing of the bank of register files when the corresponding components of each vector are stored in a register of the bank in horizontal and vertical write operations, and to control the addressing of the bank to collect the components of each vector for subsequent output rotation in horizontal and vertical read operations, said controller further configured to control alternating

2

horizontal reading and writing and vertical reading and writing operations upon the bank of register files.

2.   (Original) The data converter of claim 1,

wherein each vector has n components indexed from 0 to n-1 such that there are 0 to n-1 sets of corresponding components; and

wherein the amount of rotation by the input rotator is zero for the 0th set of corresponding components, and n-1 steps clockwise for the (n-1)th set, any intervening sets of corresponding components being rotated by an amount equal to the ordinal number of the set.

3.   (Original) The data converter of claim 1,

wherein there are n vectors indexed from 0 to n-1; and

wherein the amount of rotation by the output rotator is zero for the 0th vector and n-1 steps counter-clockwise for the (n-1)th vector, any intervening vectors being rotated by an amount equal to the ordinal number of the vector.

4.   (Original) The data converter of claim 1, wherein each register file in the bank includes a register for storing the vector components.

5.   (Original) The data converter of claim 4, wherein each vector has n components and each register file in the bank has n component registers.

6.   (Original) The data converter of claim 5, wherein there are *n* register files in the bank.

7.   (Original) The data converter of claim 1, wherein the bank of register files is configured to write and read the vector components at the same clock cycle.

8.   (Canceled)

9.   (Previously Presented) The data converter of claim 1, wherein the vector has *n* components and the controller horizontally writes *n* sets of corresponding components and horizontally reads *n* vectors.

10.   (Original) The data converter of claim 9, wherein, after the controller horizontally writes *n* sets of corresponding components and horizontally reads *n* vectors, the controller vertically writes *n* sets of corresponding components and vertically reads *n* vectors.

11.   (Original) The data converter of claim 1, wherein the output rotator rotates the vector component a position equal and opposite to the input rotator.

12.   (Previously Presented) A method for converting a group of vectors from a time serial to a time parallel format, wherein in the time serial format, sets of

corresponding components of the vectors each have a time slot, and in time parallel format, each vector has a time slot, the method comprising:

    for each set of corresponding components,

        rotating the corresponding components an amount that depends on the time slot of the corresponding component;

        writing each set of rotated corresponding components in a separate set of registers in a bank of register files; and

    for each vector in the group,

        reading selected registers in the bank to collect the components of the vector; and

        rotating the collected components of the vector an amount that depends on the time slot of the vector to generate a vector in time parallel format;

    wherein writing each set of corresponding components and reading the vector components is performed either horizontally or vertically, in an alternating fashion.

13.   (Original) The method of claim 12, wherein if the vector components are written horizontally to the bank of register files, then the vector components are read horizontally from the bank of register files.

14.   (Original) The method of claim 12, wherein if the vector components are written vertically to the bank of register files, then the vector components are read vertically from the bank of register files.

15.   (Original) The method of claim 12, wherein a set of corresponding components is written and the components of a vector are read in the same clock cycle.

16.   (Original) The method of claim 12,

wherein the vector has n components;

and wherein n sets of corresponding components are horizontally written over n clock cycles and vectors are horizontally read over the same n clock cycles.

17.   (Original) The method of claim 16, wherein in another n clock cycles subsequent to the n clock cycles, n sets of corresponding components are vertically written over n clock cycles and vectors are vertically read over the same n clock cycles.

18.   (Currently Amended) A[[n]] data converter for converting a group of vectors from a time serial to a time parallel format, wherein in the time serial format, sets of corresponding components of the vectors each have a time slot, and in time parallel format, each vector has a time slot, the converter comprising:

input rotation means for rotating each set of corresponding components of all vectors by a first prescribed amount depending on the particular set;

storage means coupled to the input rotation means, for storing the rotated set of corresponding components; and

output rotation means coupled to the storage means, for receiving components of a vector from the storage means and rotating the components of the vector by a second

6

prescribed amount depending on the particular vector to generate a vector in time parallel format; and

controller means, communicably coupled to the input rotator means, the storage means and the output rotator means, for controlling the writing and reading of the vector components to the storage means and the rotation of the vector components by the output rotation means and the input rotation means, and for controlling said reading and writing operations horizontally and vertically, said horizontal operations alternating with said vertical operations.

19.   (Original) The data converter of claim 18 wherein:

the input rotation means is an input rotator configured to rotate each set of corresponding components of all vectors by an amount that depends on the time slot of the set of corresponding components;

the storage means is a bank of register files with a register file in the bank configured to store each rotated set of corresponding components; and

the output rotation means is an output rotator configured to receive and rotate the components of a vector an amount that depends on the time slot of the vector.

20.   (Original) The data converter of claim 19 wherein the storage means is configured to write and read the vector components in the same clock cycle.

21.   (Original) The data converter of claim 20 wherein the storage means is configured to write corresponding components horizontally and then read vectors horizontally over a prescribed number of clock cycles.

22.   (Original) The data converter of claim 21, wherein, during another prescribed number of clock cycles, the storage means is configured to write corresponding components vertically and then read vectors vertically.

23. – 24.   (Canceled)

25.   (Original) The data converter of claim 18 wherein the output rotation means rotates time parallel vector components in a direction opposite to the direction that the input rotation means rotates a set of corresponding vector components.